

AMENDMENTS TO THE CLAIMS

Please amend the claims as indicated below. The language being added is underlined (“ ”) and the language being deleted contains a strikethrough (“”).

LISTING OF CLAIMS

1-2. (Canceled)

3. (Previously Presented) A method of performing cubic mapping as recited in claim 11, wherein the continuity adjustment code is obtained from a table of codes, the table being indexed by the face ids.

4. (Previously Presented) A method of performing cubic mapping as recited in claim 11,

wherein the approximation to the derivative of the texture coordinates is based on the difference between the texture coordinates of the neighboring pixels; and

wherein updating the current texture axes includes swapping orientation of the current texture axes of one of the faces.

5. (Previously Presented) A method of performing cubic mapping as recited in claim 4,

further comprising the step of normalizing the texture coordinates prior to computing the LOD parameter; and

wherein updating the current texture axes includes compensating for the normalizing step.

6. (Original) A method of performing cubic mapping as recited in claim 5, wherein the step of compensating includes adding or subtracting one.

7. (Previously Presented) A method of performing cubic mapping as recited in claim 11,

wherein the approximation to the derivative of the texture coordinates is based on the difference between the texture coordinates of the neighboring pixels; and

wherein updating the current texture axes includes flipping the orientation of one of the current texture axes of one of the faces.

8. (Previously Presented) A method of performing cubic mapping as recited in claim 7,

further comprising the step of normalizing the texture coordinates prior to computing the LOD parameter; and

wherein updating the current texture axes includes compensating for the normalizing step.

9. (Original) A method of performing cubic mapping as recited in claim 8, wherein the step of compensating includes adding or subtracting one.

10. (Currently Amended) A method of performing cubic mapping with texturing, comprising:

selecting neighboring pixels to be mapped;

mapping each selected pixel to a cube face based on the magnitude of a normal vector for each face;

for each of the faces to which the selected pixels are mapped, assigning a face id and defining the orientations of the texture axes of the faces in terms of the normal vectors of the cube faces, said texture axes being current texture axes;

if the neighboring pixels do not have the same face id, changing the current texture axes of at least one of the faces based on assigned face ids to make the current texture axes consistent and continuous between the faces;

computing an approximation to [[the]] derivatives of the texture coordinates based on the current texture axes; and

computing, based on continuity-adjusted derivatives, a level of detail (LOD) parameter for the texturing of said pixels.

11. (Currently Amended) A method of performing cubic mapping as recited in claim 10, wherein the step of changing the current texture axes based on the assigned face ids:

obtaining a continuity adjustment code based on the assigned face ids; and
updating the current texture axes of at least one of the faces based on the adjustment code to make the current texture axes consistent and continuous between the faces.

12. (Currently Amended) A program storage device readable by a machine tangibly embodying a program of instructions executable by a machine to perform a method of cubic mapping with texturing, said method comprising:

selecting neighboring pixels to be mapped;
mapping each selected pixel to a cube face based on the magnitude of a normal vector for each face;
for each of the faces to which the selected pixels are mapped, assigning a face id and defining the orientations of the texture axes of the faces in terms of the normal vectors of the cube faces, said texture axes being current texture axes:
if the neighboring pixels do not have the same face id, changing the current texture axes of at least one of the faces based on assigned face ids to make the current texture axes consistent and continuous between the faces; and
computing an approximation to [[the]] derivatives of the texture coordinates based on the current texture axes; and

computing, based on the approximations of the derivatives, a level of detail (LOD) parameter for the texturing of said pixels.

13. (Currently Amended) The ~~method~~program storage device of ~~performing cubic mapping as recited in~~ claim 12, wherein the step of changing the current texture axes based on the assigned face ids includes:

obtaining a continuity adjustment code based on the assigned face ids;

updating the current texture axes of at least one of the faces based on the adjustment code to make the current texture axes consistent and continuous between the faces;

14. (Currently Amended) The ~~method~~program storage device of ~~performing cubic mapping as recited in~~ claim 13, wherein the continuity adjustment code is obtained from a table of codes, the table being indexed by the face ids.

15. (Currently Amended) The ~~method~~program storage device of ~~performing cubic mapping as recited in~~ claim 13,

wherein the approximation to the derivative of the texture coordinates is based on the difference between the texture coordinates of the neighboring pixels; and

wherein updating the current texture axes includes swapping orientation of the current texture axes of one of the faces.

16. (Currently Amended) The ~~method~~program storage device of ~~performing~~
~~cubic mapping as recited in~~ claim 15,

further comprising the step of normalizing the texture coordinates prior to
computing the LOD parameter; and

wherein updating the current texture axes includes compensating for the
normalizing step.

17. (Currently Amended) The ~~method~~program storage device of ~~performing~~
~~cubic mapping as recited in~~ claim 16, wherein the step of compensating includes adding
or subtracting one.

18. (Currently Amended) The ~~method~~program storage device of ~~performing~~
~~cubic mapping as recited in~~ claim 13,

wherein the approximation to the derivative of the texture coordinates is based
on the difference between the texture coordinates of the neighboring pixels; and

wherein updating the current texture axes includes flipping the orientation
of one of the current texture axes of one of the faces.

19. (Currently Amended) The ~~method~~program storage device of performing ~~cubic mapping as recited in~~ claim 18,

further comprising the step of normalizing the texture coordinates prior to computing the LOD parameter; and

wherein updating the current texture axes includes compensating for the normalizing step.

20. (Currently Amended) The ~~method~~program storage device of performing ~~cubic mapping as recited in~~ claim 19, wherein the step of compensating includes adding or subtracting one.